



Design for Improve Traffic Reliability in Overlay Multicast Networks

Chung-Ryeon Kim, Sung-Uk Choi*

Department of Computer Science and Engineering, Incheon National University

ABSTRACT

As for overlay multicast which provides media streaming, it should cope with dynamic changes in nodes rapidly, and requires methods for error-recovery that take into data bandwidth that each node can process and time during which the node stay on the system. Therefore, this paper uses the characteristics of Bandwidth-Ordered Tree and Time-Ordered Tree for constructing a stable tree in a dynamic environment, and suggests Stay-Time Tree that has Stay Index for efficient error recovery. This algorithm is a method that forms the recovery group, considering the characteristics and time during which each of nodes stays on the system, so that it may be applied to wire & wireless P2P network in an integrated manner. It was found that recovery using this method improved communication reliability about 20%, compared to the existing method.

© 2014 KKITS All rights reserved

KEYWORDS : Overlay multicast, Multicast tree, Multicast service, Distributed streaming, P2P

ARTICLE INFO: Received 24 January 2014, Revised 11 February 2014, Accepted 14 February 2014.

1. 서론

*Corresponding author is with the Department of the Computer Science and Engineering, Incheon National University, 119 Academy-ro Yeonsu-Gu Incheon, 407-772, KOREA

E-mail address: swchoi@incheon.ac.kr

오버레이 멀티캐스트는 IP 멀티캐스트가 가지고 있는 여러 문제점을 해결하는 측면에서 널리 사용되고 있다. 이러한 환경에서는 노드들의 여건 뿐 만 아니라 효율적인 오류 복구 방법도 노드에서 제공되어야 한다[1]. 또한 현재 유무선 노드들이 혼재된 P2P 네트워크에서는 무선 디바이스를 위한 정책도 고려되어야한다. <그림 1>은 모바일

및 유선 디바이스들이 혼합된 통합 오버레이 P2P 네트워크이다. 먼저 노드들의 동적인 환경에 대응하여 안정적인 트리를 구성하기 위한 알고리즘으로 대표적인 것으로 Bandwidth-Ordered (BO)[2] 트리와 Time- Ordered (TO)[2] 트리를 들 수 있으며, 데이터 복구 방법에 있어서도 부모노드에게 요청하는 Parent- recovery[3], 서버에게 요청하는 Source- recovery [3]와 같은 많은 방법들이 있다.

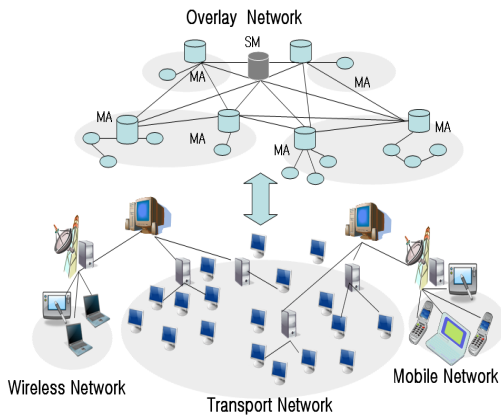


그림 1 통합 P2P 오버레이 네트워크
Figure 1. Integrated P2P Overlay Network

본 논문에서는 P2P 멀티캐스트 환경을 위하여 Stay-Time(ST) 트리를 제안하는데, BO 트리나 TO 트리가 가지고 있는 장점들을 이용해서 유무선 디바이스의 특성에 적합한 오류복구 방법을 제안하고자 한다. 기존의 모바일 P2P 네트워크 정책은 유선 P2P를 기본으로 하여 일부 기능을 추가하거나 혹은 기존 기능을 확장한 개념이 대부분이라 할 수가 있다. [4, 5] 그러나 모바일 디바이스는 가용 자원 및 서비스 시간의 제약성 및 이동성의 특성이 있기 때문에 이러한 특성을 감안한 스케줄링이 필요하다. 본 논문 2장에서는 BO 트리와 TO트리의 관련연구와 오버레이 멀티캐스

트에서 데이터를 복구하는 방법에 대한 소개를 하고, 3장에서 제안한 안정적인 오류복구 방식을 설명한다. 4장에서는 본 제안 알고리즘을 기반으로 하는 트래픽 에이전트를 포함하는 미들웨어를 제안한다. 5장에서는 실험과 성능 평가를 하여 유효성을 확인하며, 6장에서는 실험결과를 바탕으로 결론을 맺는다.

2. 관련연구

2.1 안정적인 트리 구성 알고리즘

노드들의 동적인 환경에서 안정적인 트리를 구성하기 위해서 많은 오버레이 멀티캐스트 알고리즘이 제안되었으며[1][6][7]. 대표적으로 Bandwidth-Ordered(BO) 트리와 Time-Ordered (TO) 트리가 있다. 각 노드들의 대역폭에 따라 순차적으로 구성하는 것을 BO 트리라고 한다. 대역폭에 따라 구성했기 때문에 트리의 깊이가 상대적으로 짧게 구성되며, 노드들의 동적인 상황에서도 영향을 적게 받게 된다. 반면에, TO 트리는 노드들이 시스템에 조인(join)된 시간에 따라 트리를 구성하는 것이다. 전체적인 트리는 시스템에 머무른 시간에 따라 정렬되는데 이것을 TO 트리라고 한다.

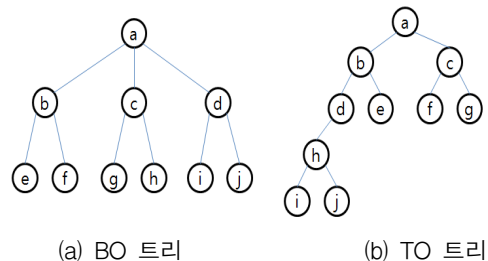


그림 2. 노드의 동적 환경에 따른 트리구성 방법
Figure 2. Tree Architecture in Dynamic Environment of Nodes

표 1. BO 트리와 TO 트리의 비교
Table 1. The Comparison of BO and TO

트리	비교
Bandwidth - Ordered Tree (BO)	- 대역폭을 너무 오래 유지하거나 너무 크게 유지하면 노드들 간 대역폭 불균형을 초래 함 - 대역폭이 큰 노드가 머무르는 시간이 짧은 경우 많은 하위노드들의 소멸이 발생할 수 있으며 복구 시간이 오래 걸림 - Service delay : TO에 비해 낮음 - Overhead : TO에 비해 낮음 - starving Time : TO에 비해 낮음
Time - Ordered Tree (TO)	- 트리의 깊이가 너무 긴 경우 노드들의 동적인 상황에 안정적으로 대처하지 못함 - 따라서 실시간 스트리밍 서비스에서 오버헤드나 딜레이가 길어 질 수 있음 - Service delay : BO에 비해 높음 - Overhead : BO에 비해 높음 - starving Time : BO에 비해 높음

BO트리의 동적 트리구성 방법은 order statistics[8] 알고리즘에 기반한다. 트리의 레이어에서의 피어의 수를 X_1, X_2, \dots, X_n 로 정의하고, 이는 확률밀도 함수(PDF : Probability Density Function) $f_X(x)$ 와 누적확률분포함수(CDF : Cumulative - probability Distribution Function) $F_X(x)$ 를 따른다고 가정한다. 이를 수학적으로 표현하면,

$$S = \{X_i | a < X_i < b\} \quad \text{이 고,} \quad \bar{S} = \frac{1}{|S|} \sum_{X_i \in S} X_i$$

일 때, $\bar{S} \rightarrow \frac{1}{F_X(b) - F_X(a)} \int_a^b f_X(x) x dx$ 와 같이 나타낼 수 있다. TO트리에서 전체 피어(peer)의 수는 TO의 대역폭에 의해 결정된다고 가정하고, 각 레이어에서의 피어의 합과 같다. 즉, $X_{k+1} = B_k = \sum_{j=1}^{X_k} \xi_j$ 이다. 트리의 동적 트리구성 방법은 마코프체인 알고리즘에 기반한 branching process[10] 로서 이를 수학적으로 표현하면,

$$E[X_k] = C\mu_\xi^{k-1} \quad \text{과}$$

$$Var[X_k] = \begin{cases} kC\sigma_\xi^2 & \text{if } \mu_\xi = 1 \\ \frac{(\mu_\xi^{k-1} - 1)\mu_\xi^{k-2}C}{\mu_\xi - 1} \sigma_\xi^2 & \text{if } \mu_\xi \neq 1 \end{cases}$$

와 같이 나타낼 수 있다. (X : 피어의 수, C : 상위 노드의 대역폭, ξ : 각 피어의 대역폭)

2.2 데이터 복구 알고리즘

오버레이 멀티캐스트에서 데이터를 복구하는 방법으로 많은 알고리즘이 제안되었다. 데이터 복구 방법으로 크게 proactive 방식과 reactive 방식으로 나눌 수 있으며 <표 2>과 같다[9][10].

표 2. 오버레이 멀티캐스트에서의 데이터 복구방법
Table 2. Data Recovery in Overlay Multicast

방법	종류 및 특징
Proactive	<ul style="list-style-type: none"> • Erasure-resilient coding • Probabilistic-resilient coding • Multiple-path transmission
	특징 : 오류가 발생하기 전에 미리 중복된 데이터 패킷을 보냄으로써 데이터 전송 효율을 높이고 빠르게 복구할 수 있다.
	<ul style="list-style-type: none"> • Source-recovery • Parent-recovery • Recovery-neighbor
Reactive	특징 : 오류가 발생하면 재전송을 요청하는 방식으로 복구는 느리지만 오버헤드는 적다

본 논문에서는 Reactive의 Recovery-neighbor 알고리즘과 유사한 방식을 사용하며, 재전송을 요청할 노드들로 구성된 복구 그룹을 미리 만든다. 패킷오류가 발생하면 미리 구성된 복구 그룹을 통해서 데이터를 재전송 받는다.

또한 Parent-recovery나 Source-recovery에서는 어떤 한 노드한테만 데이터 재전송을 요청하기 때문에 그 노드만으로 재전송이 집중되는 문제가 발생할 수도 있다. 따라서 제안하고자 하는 Stay-Time(ST)트리는 <표 1>에서 처럼 BO 트리나 TO 트리가 가지고 있는 장점들을 이용해서 노드들의 동적인 변화에서도 안정적인 오류복구 방법을 제시하고자 한다.

3. 제안된 안정적인 오류복구 방식

3.1 트리 형성

트리 구성은 미디어 스트리밍 서비스를 위해서 BO트리 방식을 따른다. 즉, 서비스 딜레이를 줄이고 데이터 효율을 높이기 위해서 트리의 깊이를 짧게 구성하는 것이다. 노드가 시스템에 참여하게 되면 서버는 트리의 상위부터 검색해서 들어갈 위치를 알려준다. 그러면 노드는 그 위치에 들어가게 되고 남겨진 노드가 있으면 재구성하게 된다.

노드 V 와 에지 E 를 가지는 트리 $T=(V,E)$ 가 있다고 가정할 때, 손실상관함수 $w: V \times V \rightarrow I$ 를 정의한다. I 는 음수가 아닌 정수이고 $w(v_1, v_2)$ 는 루트 r 로부터 v_1 과 v_2 까지의 트리 경로 사이의 공통 에지의 개수(깊이)를 나타낸다. 트리 T 의 i 번째 레벨을 노드 집합 L_i 라고 하고 루트 노드 r 로만 구성된 노드 집합을 L_0 라고 가정한다. 각각의 노드는 자식노드 집합 $C_i = \{c_{i0}, c_{i1}, \dots\}$ 과 더 하위의 자손노드 집합 $D_i = \{d_{i0}, d_{i1}, \dots\}$ 을 가진다. 다음의 3.2절에서 제안된 알고리즘은 먼저 K 서브트리의 상위노드의 집합 G' 를 식별하는 것으로부터 시작한다.

ST트리에서는 데이터 복구 방법으로서 시간으로 정렬되어 있는 인덱스를 바탕으로 이루어지기 때문에 안정성이 높다고 판단된 노드에게 요청한다. 오버레이 멀티캐스트에서는 노드들의 잦은 출입으로 재전송을 요청받을 노드가 반드시 재전송을 해줄 수 있다는 것을 보장할 수 없다. 그렇기 때문에 이 인덱스에서 높은 값을 가지고 있는 노드에게 재전송을 요청하는 것이 데이터 전송 효율을 높일 수 있다.

따라서 ST트리는 대역폭에 따라 구성하되 트리의 깊이를 짧게 구성하고, 오류복구 방법으로는 노드의 머무른 시간을 이용하는 방식을 사용함으로

써 노드의 잦은 출입으로 발생하는 영향을 최소화하는 것이 목표이며, 이 기종간의 디바이스별 특성을 고려하여 부모노드와 지식노드를 검사하여 성능이 우수한 자식노드와 열세인 부모노드를 교체하는 정책을 이용하여 트리를 재배치함으로써 전체 시스템의 성능을 향상시킨다.

3.2 오류 복구 정책

시스템에 참여한 노드는 데이터를 받다가 패킷 오류가 발생하거나 부모 노드가 시스템을 갑자기 빠져 나가는 경우에 데이터를 못 받게 되면 예비 부모 노드 중에 하나에 데이터 재전송을 요청하게 된다. 실제로 데이터를 받다가 부모노드가 시스템에서 빠져 나가게 됨에 따라 자식 노드들은 데이터를 받지 못하는 경우가 있다. 자식 노드는 트리 알고리즘에 따라 트리를 재구성해야 하고 다시 데이터를 받게 된다. 트리를 재구성하는데 시간이 오래 걸리기 때문에 그 동안 복구 그룹의 노드로부터 데이터를 잠시 동안 받게 된다.

ST트리의 기본구성 방식은 BO트리의 구성방식을 따른다. 미디어 스트리밍의 특성상 서버에서 전송되어지는 데이터는 딜레이가 짧은 것이 데이터 효율을 높일 수 있을 것이다. 그리고 오류복구 방법으로는 ST트리의 Stay Index 라는 것을 바탕으로 오류복구를 시도한다. Stay Index는 전체 시스템에 참여하고 있는 노드가 시스템에 들어와서 머무른 시간으로 정렬되어 있는 데이터 구조라고 할 수 있다.

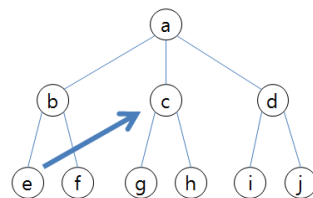


그림 3. 오버레이 멀티캐스트 트리
Figure 3. A tree of Overlay Multicast

ST트리에서는 복구 그룹을 형성하는 데 있어 노드들이 얼마나 신뢰성 있는지를 판단하기 위해서 ST트리에 Stay index를 부여하고 이를 이용한다. <그림 3>은 오버레이 멀티캐스트 트리의 한 부분으로서 대역폭 속성만 고려했기 때문에 상위 노드는 하위노드보다 대역폭이 크게 구성된다.

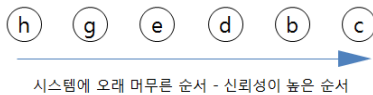


그림 4. Stay - Time의 Stay Index 부여
Figure 4. Assign of Stay Index for Stay - Time

<그림 4>는 각 노드들이 시스템에 들어온 시간으로 정렬된 것이다. 상위에 있는 노드가 하위에 있는 것보다 시스템에 머무른 시간이 더 길다는 것을 의미한다.

<그림 3>에서 노드 e는 자기의 복구 그룹으로써 노드 c가 포함되어 있다. 노드 e는 복구 그룹에 들어갈 노드들을 자기의 상위 노드 중에서 선택하게 된다. 상위 노드가 자신보다 데이터를 빨리 받을 확률이 더 높기 때문에 상위 노드를 복구 그룹 후보노드(노드 b, c, d)로 선정하게 된다. 이들 중에 <그림 4>의 Stay Index 로 보았을 때 노드 c가 인덱스 상에서 가장 큰 값을 가지는 것을 알 수 있다. 따라서 노드 e의 복구 그룹에 포함되는 것은 노드 c가 선택된다.

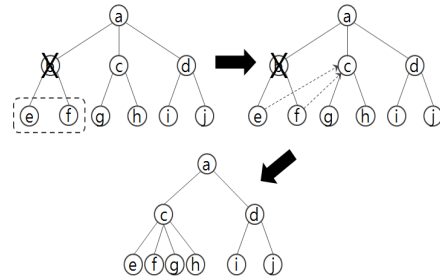
<그림 5>는 노드 b가 시스템으로부터 소멸하는 경우 이의 복구 과정을 나타내고 있다.

이를 알고리즘으로 표현하면 다음과 같다.

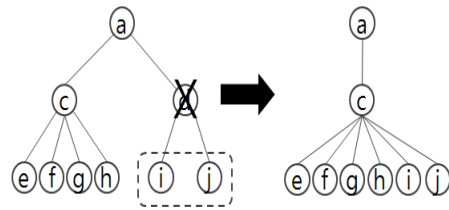
1. 상위 노드 집합 초기화 : $G' \leftarrow 0$
2. T에서 L_0 로부터 시작하고 첫 번째 레벨 L_i 을 발견한다. 즉, $|L_i| < K \leq |L_{i+1}|$
3. 각 $v_i \in L_i$ 이고, v_i 의 임의의 자식노드를 취하여 c_{ij} 라고 둔다. $|G'| \geq K$ 될 때까지

$G' \leftarrow G' + \{c_{ij}\}$ 이고 $G \leftarrow G - \{c_{ij}\}$ 이다.

4. 각 $v_i \in G'$ 이고, 임의의 더 하위 자식노드 $d_{ij} \in D_i$ 이고 $G \leftarrow G + \{d_{ij}\}$ 이고 $D_i \leftarrow D_i - \{d_{ij}\}$ 이다.



(a) 노드 b가 사라지는 경우 각 복구그룹 c, d 정보에 의한 노드 e 복구 및 f 복구



(b) 노드 d가 사라지는 경우 복구그룹 c를 이용한 복구

그림 5. 소멸된 노드 하위 노드의 복구 과정
Figure 5. Recovery Process of destroyed nodes of child nodes

<그림 5> (a)에서 노드 e, f의 복구가 완료된 이후 (b)에서 노드 d가 삭제된다면 Stay Index 내에 머무른 시간 순으로 볼 때 노드 c를 복구 그룹으로 결정한다.

3.3 통합된 P2P 복구 그룹 재형성

P2P 서비스에 모바일 노드가 참여하는 경우에는, 모바일 P2P는 단말기의 특성 및 환경, 멀티미디어 콘텐츠 서비스의 특징을 모두 수용하여야 한다. 따라서 이동 및 서비스 범위, 기기의 처리 능력의 제한 등의 이유로 유선 P2P 노드의 서비스

정책과 차이를 두어야 할 필요가 있다. 예를 들어 유선 P2P에서는 동일한 P2P 세션에 조인된 시간이 오래될수록 노드의 신뢰성이 있다고 판단하였지만, 모바일 디바이스는 배터리 용량 및 가용 자원의 한계성으로 세션에 조인된 시간이 오래될수록 해당 P2P 트리를 떠날 가능성이 높다고 할 수 있다. 그 외에 가용 대역폭, 무선 트래픽 비용 등의 요인이 상대적으로 중요하다. 따라서 일반적인 Stay Index만으로 복구그룹을 형성할 경우, 이들 부모노드와 자식노드간의 대역폭과 P2P네트워크상에서 디바이스 특성을 무시한 정책은 전체 시스템의 성능을 보장하지 못한다. 따라서 노드 i 의 최대 가용대역폭을 bw_i^{\max} , 전송비용을 ct_i , 현재시간을 cut_i , 처음 트리에 조인된 시간 jnt_i , 시간 요인 fct_i 라 하면, 통합 P2P의 Stay Index의 후보순위를 결정하기 위한 산출 값은 $fct_i \times (\frac{bw_i^{\max}}{ct_i})$ 이 된다.

여기서 시간 요인은 유선 노드일 경우 $fct_i = cut_i - jnt_i$ 이 되고, 모바일 노드일 경우는 $fct_i = \frac{1}{cut_i - jnt_i}$ 이 된다.

또한 Stay Index 후보순위가 결정되면, 다음으로 자식노드들의 진입을 허용할 수 있는 대역폭인가를 조사해야 한다. 따라서 다음의 수식을 통하여 자식노드의 진입가능 여부를 조사한다.

$$Outdeg_i^{avail} = \left(\frac{\alpha \times bw_i^{\max}}{bw^{request} + bw^{request-1}} \right)$$

여기에서,

- $Outdeg_i^{avail}$: 노드 i 가 가질 수 있는 대역폭을 감안한 총 자식노드의 수

- α : 노드 i 의 트래픽 비용에 대한 가중치이며 일반적인 경우 1이지만, 무선노드일 경우 통신 방식에 따라 $0.1 \leq \alpha \leq 1$ 의 값을 지정할 수 있다.

- bw_i^{\max} : 노드 i 의 최대 가용대역폭
- $bw^{request}$: 노드 i 에 새롭게 요청하는 자식노드 대역폭의 합
- $bw^{request-1}$: 노드 i 에 현재 서비스를 받고 있는 자식노드 대역폭의 합

<그림6>은 노드(c)의 $Outdeg_i^{avail}$ 의 수가 1보다 작으면, 대역폭이 부족한 관계로 노드 (i)(j)는 통합 P2P 후보 Stay Index에서 다음순위 노드의 자식으로 이동시킨 결과를 보여준다.

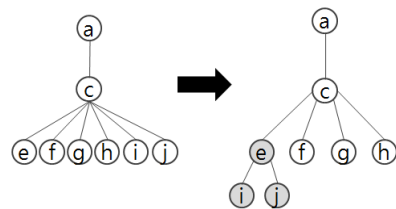


그림 6. 대역폭에 따른 트리의 재구성
Figure 6. Reconstruction of the tree due to the bandwidth

4. 트래픽 에이전트 기반 미들웨어 설계

4.1 트래픽 에이전트

트래픽 에이전트는 클라이언트간의 패킷 대역폭을 체크하여 토폴로지 상의 최적화된 상위 클라이언트를 찾는 것이며 하위노드 컨트롤 모니터링을 통해서 자신에게 접속되어 있는 클라이언트의 정보를 확인하는 기능을 가진다. 다음으로는 세션 컨트롤러로 모든 클라이언트의 활성화 여부를 확인하는 기능을 담당한다. 마지막으로 노드 컨트롤러로 클라이언트간의 연결을 담당한다.

데이터베이스에 저장되는 값으로는 각 상위 IP와 현재 기준 IP 그리고 하위 IP의 각각의 속도 정보, 그리고 누적된 노드 수가 있다. <그림 7>에서 트래픽 에이전트의 동작을 보면 처음 클라이언트

가 접속해서 자신이 최상위 클라이언트일 경우에는 메인 서버에 접속되며 그 이후에 접속하는 클라이언트는 에이전트의 통제 하에 각각 상위 클라이언트와 자신과의 속도를 체크한다.

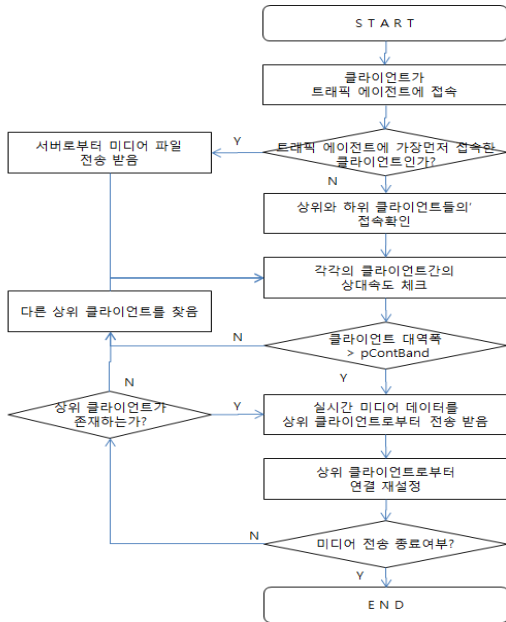


그림 7. 트래픽 에이전트 프로세스
Figure 7. Traffic agent process

즉 제어 파라미터 값과 비교하여 서비스 가능 여부를 확인하게 된다. 이 때 식 1에서 제어 파라미터 값인 $pContBand$ 는 콘텐츠의 크기를 상위 클라이언트가 현재 서비스해주고 있는 노드수로 나눈 것이다.

5. 실험 및 성능평가

5.1 성능분석 환경

성능분석 환경은 쉘넷 5.1[11] 환경에서 오버레이 멀티캐스트 환경을 구현하였다. 오버레이 네트워크 구성은 Georgia Tech Internetwork Topology Model (GT-IMT) Transit-stub 모델을 이용하여 구

성하였다. 전송비용과 가용대역폭을 달리한 1200개의 노드를 생성하였고 각 Stub 노드는 하나의 사용자 단말이 붙어 있는 것을 가정하였다. 통합 P2P 네트워크 환경을 위하여 총 노드 중 20%는 모바일 노드로 하고, 이를 관리하는 별도의 모바일 슈퍼노드에 연결하였다. 모바일 노드는 조인 이후 멀티미디어 스트림을 다운 받는 시간(360 초)이후는 해당 세션을 떠나는 것으로 가정하였다. 노드의 가용 대역폭의 분포와 시스템에 머무르는 시간의 분포는 Log-Normal 분포를 따른다. 노드가 시스템에 머무르는 시간, Life-Normal의 분포 $LN(10,2)$ 는 평균 600초를 시스템에 머무르는 것으로 설정했다. 실험 대상으로는 BO 트리, TO 트리, 제한한 ST 트리 방식 등 총 3가지의 오버레이 멀티캐스트 트리 알고리즘을 구현하여 실험하였다.

5.2 서비스딜레이

<그림 8>은 노드의 라이프타임에 따라 각 알고리즘의 평균 서비스 딜레이를 나타낸다. 서버에서 서비스를 받는 각 사용자 노드까지 평균 걸리는 시간을 의미한다.

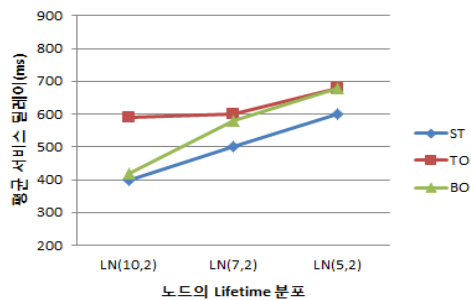


그림 8. 노드의 변화에 따른 서비스 딜레이
Figure 8 Service delays due to changes in node

노드들의 라이프타임이 짧아질수록 노드들의 평균 딜레이는 전체적으로 커진다. 즉 시스템에서 노

드들이 오래 머무르지 못하고 잦은 출입이 있을수록 트리의 재구성이나 패킷 전송 효율이 떨어지기 때문에 딜레이는 길어진다.

BO트리와 ST트리는 대역폭에 따라 정렬되기 때문에 서비스 딜레이 측면에서는 TO 트리보다 더 좋은 결과를 보인다. 시스템에서 노드들의 출입이 잦게 되면 위의 그래프에서 보듯이 전체적으로 서비스 딜레이가 커지는 것을 알 수 있다. 라이프타임을 고려했기 때문에 TO 트리는 그 차이가 심하지 않지만 BO 트리의 경우에는 그 차이가 많이 나는 것을 알 수 있다. ST트리는 평균적인 서비스 딜레이도 낮고 노드들의 변화가 심한 경우에도 복구 방법으로 라이프타임을 고려하기 때문에 어느 정도 안정적인 결과를 보이는 것을 알 수 있다.

5.3 오버헤드

<그림 9>는 각 알고리즘에 따라 발생하는 컨트롤 패킷의 수를 나타낸다.

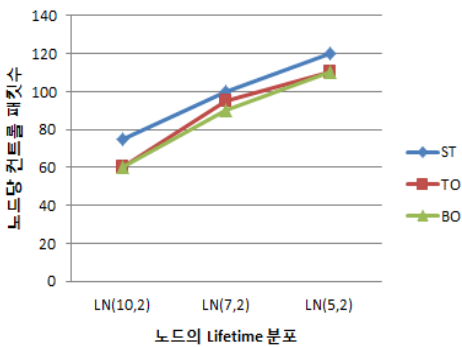


그림 9. 노드의 변화에 따른 오버헤드
Figure 9. Overhead due to changes in node

각 알고리즘들은 대역폭이나 시간을 고려해서 트리를 구성하기 때문에 특별한 방법을 사용하지 않는 경우보다는 컨트롤 패킷의 수가 많은 편이다. 즉, 노드들의 동적인 변화가 많은 상황에서는 전

체적으로 컨트롤 패킷의 수가 많아진다. ST 트리는 데이터 복구 방법으로 Stay index를 이용하기 때문에 더 많은 작업이 필요하다. 그래서 전체적으로 다른 알고리즘보다 컨트롤 패킷의 수가 많아진다.

5.4 Startup Latency

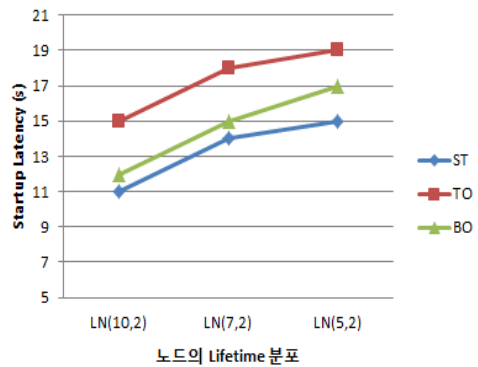


그림 10. Startup-Latency에 따른 전송 데이터 효율
Figure 10. Efficiency of data transmission due to Startup-Latency

<그림 10>은 각 트리의 Startup-Latency를 나타낸 것이다. Startup-Latency는 노드가 시스템에 참여해서 10초 분량의 데이터를 받기 까지 걸리는 시간을 나타낸다. 시스템에 참여해서 초기에 재생할 수 있는 분량의 데이터를 받기까지 걸리는 시간은 실시간 스트리밍 서비스에 있어서 노드의 탈퇴율에도 영향을 미치기 때문에 중요한 요소이다.

서비스 딜레이와 마찬가지로 네트워크 크기가 증가함에 따라 Startup-Latency가 증가하는 것을 볼 수 있다. TO인 경우 데이터를 얻어오는 시간이 길기 때문에 BO보다 지연이 더 긴 것을 알 수 있다. 이 논문에서 제안한 방법도 BO처럼 대역폭을 고려해서 트리를 만들기 때문에 Startup-Latency가 다른 트리보다 적은 것을 확인 할 수 있다.

6. 결 론

미디어 스트리밍을 위한 오버레이 멀티캐스트에서는 노드들의 잦은 출입에 영향을 적게 받고 효율적인 데이터 전송 방법이 필요하다. 본 논문에서 제시한 ST트리는 전송 딜레이를 줄이기 위해서 트리의 깊이를 짧게 구성하면서 노드들의 잦은 출입으로 영향 받는 것을 최소화하기 위해서 데이터 복구 방법으로 노드들의 라이프타임과 가용대역폭 및 전송 비용을 고려한 통합 P2P 오버레이 네트워크를 고려해서 복구 그룹을 형성하였다. 그 결과 데이터 전송 딜레이 측면에서도 높은 결과를 얻을 수 있고 노드들의 잦은 출입으로 받는 영향도 다른 알고리즘에 비해 높은 결과를 얻을 수 있었다.

또한, 노드의 동적인 변화가 많아지면 상위노드들로부터 데이터를 못 받는 시간이 길어진다. 이 때 효과적으로 빠르게 트리를 재구성하여 데이터를 전송받는 것이 중요하다. ST 트리는 데이터 복구 그룹을 유지하면서 데이터를 재전송하기 때문에 전체적으로 데이터를 받는 효율이 높아진다. 또한 모바일 노드일 경우에도 다른 알고리즘과는 달리 좀 더 안정적으로 데이터를 받을 수 있다.

References

- [1] JARVIS et al, *Performance analysis and improvement of overlay construction for peer-to-peer live streaming*, SIMULATION, Vol. 82, Issue 2, February, pp. 93-106, 2006
- [2] JARVIS et al, *Constructing reliable and efficient overlays for p2p live media streaming*, I.J. of SIMULATION Vol. 7 No. 2, pp. 54-62, 2006
- [3] Xing Jin, W.-P. Ken Yiu, and S.-H. Gary Chan, *Loss recovery in application-layer multi-cast*, MultiMedia, IEEE, Vol. 15 Issue 1, pp. 18-27, 2008.
- [4] Zhou, M., Ji, B., Han, K., & Xi, H., *A cooperative hybrid caching strategy for p2p mobile network*, Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation (ISCCCA-13), 2013.
- [5] Iyer, T., Hsieh, R., Rizvandi, N. B. Varghese, B. & Boreli, R. *Mobile p2p trusted on-demand video streaming*, arXiv preprint arXiv:1204.0094. 2012.
- [6] Guang Tan et al, *Improving the fault resilience of overlay multicast for media streaming*, Vol. 15, No. 3, IEEE Transactions on Networking, June 2007.
- [7] Evangelos Pournaras, Martijn Warnier and Frances M.T. Brazier, *Self-optimised tree overlays using proximity-driven self-organised agents.pdf*, Complex Intelligent Systems and Their Applications Springer Optimization and Its Applications Vol. 41, pp. 137-161, 2010.
- [8] David, H. A., and H. N. Nagaraja, *Order statistics. 3rd ed.* NewYork:Wiley-Interscience, 2003.
- [9] Xing Jin, W.-P. Ken Yiu and S.-H. Gary Chan, *Loss recovery in application-layer multicast*, IEEE Multimedia Magazine, Vol. 5, No. 1, 2008.
- [10] W.-P. Ken Yiu et al, *Lateral error recovery for media streaming in application-level multicast*, Vol. 4, IEEE Tran. on Multimedia, April 2006.
- [11] QualNet5.1 <http://www.scalable-networks.com>

오버레이 멀티캐스트 기반 통신 신뢰성 향상을 위한 연구

김충련, 최성욱

인천대학교 컴퓨터공학부

요 약

미디어 스트리밍을 제공하는 오버레이 멀티캐스트에서는 노드들의 동적인 변화에 빠르게 대처해야 하며, 각 노드가 처리할 수 있는 데이터 대역폭과 그 노드가 시스템에 머무르는 시간을 고려하는 오류복구 방법이 필요하다. 따라서 본 논문에서는 동적인 환경에서 안정적인 트리를 구성하기 위한 Bandwidth-Ordered 트리와 Time-Ordered 트리가 가지는 특성들을 이용해서 효율적인 오류복구를 위해 Stay Index를 가지는 Stay-Time 트리를 제안한다. 이 알고리즘은 유무선 P2P 네트워크에 통합적으로 적용할 수 있도록 각각의 노드들이 시스템에 머무르는 시간과 특성을 고려해서 복구 그룹을 형성하는 방식이다. 이를 이용하여 복구를 하는 경우 기존 방식보다 약 20% 정도 통신 신뢰성이 향상됨을 확인하였다.



Sung-Wook Choi is a professor in the Department of Computer Science and Engineering, Incheon National University. His research interests include Software System Design, Architecture Design, Software engineering, VOD and multimedia systems, Mobile Device Service Network.

E-mail address: swchoi@incheon.ac.kr

감사의 글

이 논문은 인천대학교 2013년도 자체연구비 지원에 의하여 연구되었음.



Chung-Ryeon Kim He is currently working toward a Ph.D. in the Department of Computer Science and Engineering at the Incheon National University. His research interests include VOD and multimedia systems, Software engineering.

E-mail address: kcr@hkt.co.kr